Spirit Level Reaction Time Tester

Group 52

Asmita Limaye	**2018B5A70881G**
Athreya Krishnan	**2018B4A70809G**
S. K. Anusri	**2018B5A70876G**
Prabodh Kant Rai	**2018B3A70748G**
Saurav Shukla	**2018B5AA0653G**
Varun Singh	**2018B5A70869G**

Date: 18.04.2021

Contents

User Requirements & Technical Specifications

Design a system that tests the reaction time of a person and displays a sobriety rating (1 to 5) on a 7 segment display.

The Technical Specifications are as follows

- 2 pushbuttons, START and STOP, used for user input
- START initiates testing with a random delay of [4, 8]s
- 9 LEDs light up successively at 50ms intervals after the random delay
- LEDs on/off configuration frozen if user presses STOP in < 8*50ms = 0.4s
- After 0.4s or a timely STOP press, reaction time is rated on a scale of 1 to 5.

Assumptions & Justifications

Justification

1. We check the STOP button throughout the random [4,8]s wait because a split-second check just before lighting the bottom LED excuses a user who's cheating by periodically tapping the STOP button.
2. Since we need only one interrupt, we use NMI directly to the microprocessor instead of an 8259.
3. After the sobriety rating is displayed, pressing the START button resets the system for a fresh reaction time test.

Assumptions

1. A user who presses STOP at any point before the first LED is lit up is cheating, not only just before the first LED is lit.
2. Button input level stabilises in <20ms.
3. The rating system we have decided to use for the display is as follows:

Number of LEDs lit(n)	Sobriety Score
1<=n<3	5
3<=n<5	4
5<=n<7	3
7<=n<9	2
n=9	1

Components used with justification wherever required

Qty	Component	Justification / Notes
1	8086	5 MHz minimum mode
1	8284	With a 15 MHz crystal oscillator for generating 5 MHz clock
1	LTS-4301 JR	Common Cathode 7 segment display (RED - 639 nm)
9	LEDs	Red
4	2716	Smallest ROM chip available is 2K and as we need to have even and odd bank and ROM is required at reset address which is at $FFFF0_H$ and 00000_H - where there is the IVT
2	6116	Smallest RAM chip available is 2 K and we need odd and even bank. We need RAM for stack and temporary storage of data
1	8253 / 8254	50 ms RTI and random number generation
1	8255	Interfacing 9 LEDs and 2 pushbuttons
2	TL3300 Series Tact Switch	Top actuated SPST-NO (Off-Mom) pushbuttons for START and STOP. Contact rating: 50 mA / 120 VDC
1	74LS139	Dual 2x4 decoder for memory & I/O mapping
3	74LS373	Tri-state Octal D-latches for demultiplexing AD0 - AD19 and latching A0 - A19
3	74LS245	Tri-state Octal Bus Transceivers for 16-bit data bus
1	74LS244	For buffering RD', WR', M/IO'
1	74LS760	Driving 7 segment display
. . .	Logic Gates	For control signals wherever needed
. . .	Resistors	For limiting currents and as pull-ups for active-low pushbuttons

Address Map

Memory Map

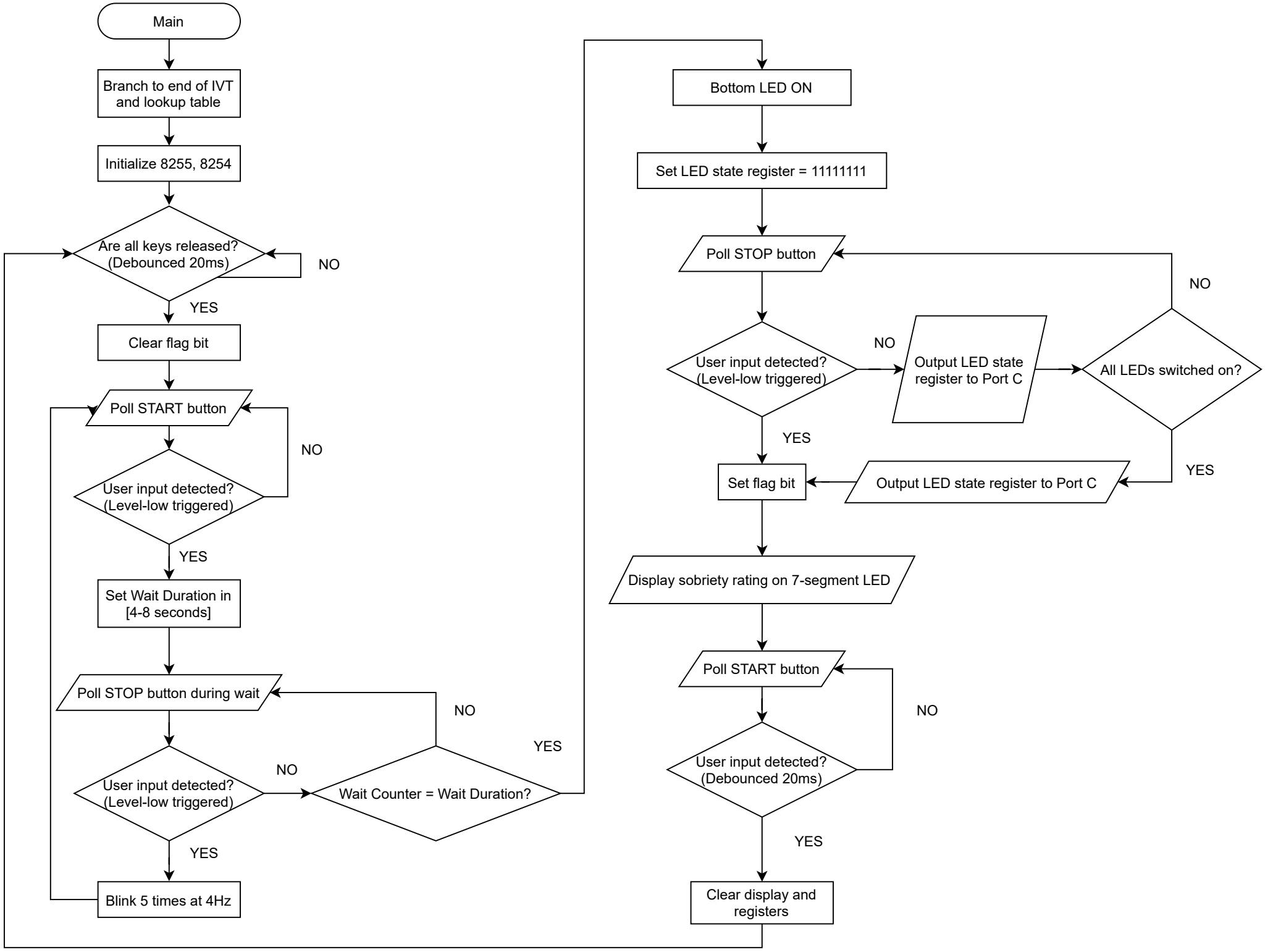
Memory	Start Address	End Address	A14	A12
ROM 1	00000_H	$00FFF_H$	0	0
RAM 1	01000_H	$01FFF_H$	0	1
ROM 2	$FF000_H$	$FFFFF_H$	1	1

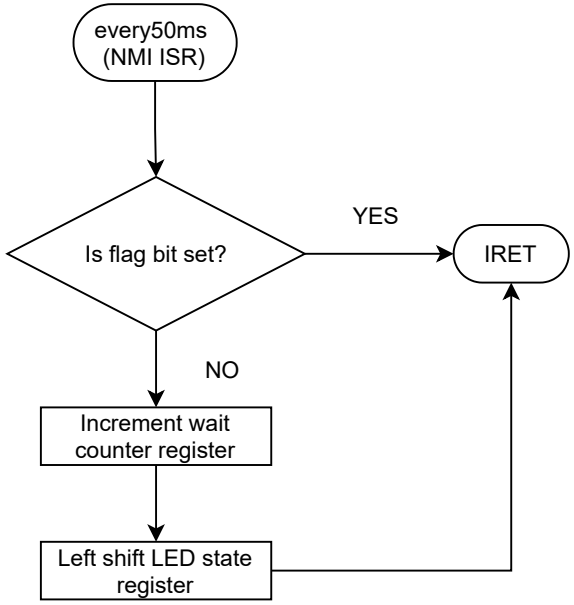
I/O Map

IO	Start Address	End Address	A3
8255	00_H	06_H	0
8253	08_H	$0E_H$	1

Design

Complete design shown with proper labelling (design attached)


Main
Branch to end of IVT and lookup table
Initialize 8255, 8254
Are all keys released? (Debounced 20ms)
NO
YES
Clear flag bit
Poll START button
NO
User input detected? (Level-low triggered)
YES
Set Wait Duration in [4-8 seconds]
Poll STOP button during wait
NO
YES
User input detected? (Level-low triggered)
NO
Wait Counter = Wait Duration?
YES
Blink 5 times at 4Hz
Bottom LED ON
Set LED state register = 11111111
Poll STOP button
NO
User input detected? (Level-low triggered)
NO
Output LED state register to Port C
All LEDs switched on?
YES
YES
Set flag bit
Output LED state register to Port C
Display sobriety rating on 7-segment LED
Poll START button
NO
User input detected? (Debounced 20ms)
YES
Clear display and registers


every50ms
(NMI ISR)
Is flag bit set?
YES
IRET
NO
Increment wait
counter register
Left shift LED state
register

Variations in Proteus Implementation with Justification

1. Memory chips excluded. Proteus simulates internal memory for 8086.
2. Using low clock frequency -- CLK = 2.4 MHz instead of 5 MHz because cascading counters to divide frequency makes simulation too slow to demonstrate.
3. Clock frequency passed in as a simulator parameter to 8086/8253 for smooth simulation.
4. Using 8253 – as 8254 not available in Proteus.
5. Used digital modeling primitive for 7 segment display which doesn't need 74LS760 as buffer.
6. Using RD'/WR' signals directly instead of IOR'/IOW'/MEMR'/MEMW'.

Firmware

Implemented using emu8086 attached. (g52_PROTEUS.asm)

List of Attachments

1. Complete Hardware Real World Design – design.pdf
2. Manuals
    a. TL3300 pushbutton
    b. S4301 JR 7 segment display
    c. 74LS760 Octal buffer and line driver
    d. 74LS139 Dual 2x4 Decoder
3. Proteus File – g52.dsn
4. EMU8086 ASM File – g52_PROTEUS.asm
5. On paper ASM File - g52_PAPER.asm
6. Binary File after assembly – g52.bin